

THE UNITED STATES PATENT AND TRADEMARK OFFICE

Appl. No. 10/796,175

Confirmation No. 3510

Inventor : ARAKAWA, H. et al.
Filed : March 10, 2004
For : REMOTE COPY SYSTEM
TC/GAU : 2161
Examiner : Safet METJAHIC
Docket No. : H-1132
Customer No.: 24956

REQUEST FOR RECONSIDERATION
OF
PETITION TO MAKE SPECIAL
UNDER 37 CFR §1.102(d) (MPEP §708.02(VIII))

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the Decision on Petition to Make Special mailed March 12, 2005, wherein the Petition was dismissed, the Applicants request reconsideration of their petition to the Commissioner to make the above-identified application special in accordance with 37 CFR §1.102(d).

I. **RESPONSE TO DISMISSAL**

The Petition was dismissed on the grounds that the discussion of the references most closely related to the invention did not include a discussion of how each of the independent claims was specifically distinguishable and patentable over

the references. Further, it was asserted that a detailed discussion of the non-patent literature document entitled *Role of Backup in Data Recovery* was not provided.

The Petition has been revised to more clearly point out to the Examiner how each of the independent claims 1 and 7 is patentable over the references deemed most closely related to the subject matter encompassed by the claims. In addition, a Second Preliminary Amendment has been filed with this Request for Reconsideration to amend these independent claims. Accordingly, the arguments for distinguishing these claims may be more concisely presented to the Examiner. Further a detailed discussion of the reference *Role of Backup in Data Recovery* has been added to distinguish the independent claims over this reference.

II. REVISED PETITION TO MAKE SPECIAL

Applicants petition the Commissioner to make the above-identified application special in accordance with 37 CFR §1.102(d). In support of this Petition, pursuant to MPEP § 708.02(VIII), Applicants state the following.

(A) REQUIRED FEE

This Petition was accompanied by the fee set forth in 37 CFR § 1.117(h) when originally filed on December 7, 2004. No additional fee is believed to be required; however, the Commissioner is hereby authorized to charge any additional payment due, or to credit any overpayment, to Deposit Account No. 50-1417.

(B) ALL CLAIMS ARE DIRECTED TO A SINGLE INVENTION

A Preliminary Amendment, in which claims 13-20 were canceled, was filed on December 7, 2004. A Second Preliminary Amendment has been filed with this Revised Petition. Claims 1 and 7 have been amended to incorporate the subject matter of claims 2 and 10, respectively, while minor changes have been made to claims 3, 4, 9, 11 and 12 for consistency purposes, and claims 2, 6, 8 and 10 have been canceled,. Accordingly, claims 1, 3-5, 7, 9, 11 and 12 are currently pending in the application. All the pending claims of the application are directed to a single invention. If the Office determines that all claims remaining in the application are not directed to a single invention, Applicant will make election without traverse as a prerequisite to the grant of special status.

As set fort in independent claims 1 and 7, the claimed invention is generally directed to a system for copying data between a plurality of storage systems. Under amended independent claim 1, the invention is a system for copying data between a plurality of storage systems, comprising: a first storage system coupled to a plurality of computers, which comprises a first logical volume storing data received from the plurality of computers; and a second storage system coupled to said first storage system, which comprises a second logical volume storing copy data of data stored in said first logical volume; wherein said plurality of computers include at least one computer issuing a write request including a write time and at least one other computer issuing a write request without a write time, when a write time is included in a write request received from one of the plurality of computers, said first storage

system records the write time and sends write data and the write time to said second storage system, when a write time is not included in the write request, said first storage system assigns a write time recorded by said first storage system to the write data and sends the write data with the write time to said second storage system, and said second storage system stores the write data received from said first storage system in said second logical volume in an order based on the write time received from said first storage system.

In addition, as recited in amended independent claim 7, the invention is a system for copying data between a plurality of storage systems, comprising: a first storage system coupled to a plurality of computers, which comprises a first logical volume storing data received from said plurality of computers; a second storage system comprising a second logical volume, which stores copy data of the data stored in said first logical volume; and a third storage system comprising a third logical volume, which stores the copy data of the data stored in said first logical volume; wherein said plurality of computers include at least one computer issuing a write request including a write time and at least one other computer issuing a write request without a write time, said first storage system stores write data received from one of said plurality of computers in said first logical volume, when a write time is included in the write request received from the one of the plurality of computers, said first storage system sends the write data and the write time to said second storage system, when a write time is not included in the write request, said first storage system sends the write data to said second storage system, when the write time and

the write data are received from said first storage system, said second storage system records the write time and sends the write data with the write time to the third storage system, when the write time is not received from said first storage system, said second storage system assigns a write time, which is recorded by said second storage system, to the write data received from said first storage system and sends the write data with the write time to said third storage system; and said third storage system stores the write data received from said second storage system in said third logical volume in accordance with the write time assigned to the write data.

(C) PRE-EXAMINATION SEARCH

A careful and thorough pre-examination search has been conducted, directed to the invention as claimed. Because the amendments to claims 1 and 7 set forth in the Second Preliminary Amendment merely incorporate the subject matter of existing claims 2 and 10, respectively, no new search is required. The pre-examination search was conducted in the following US Manual of Classification areas:

<u>Class</u>	<u>Subclass</u>
711	112, 114, 147, 154
714	6, 13

Additionally, a computer database search was conducted on the USPTO systems EAST and WEST. A keyword search was conducted on the EAST system in Class 707, subclasses 10, 102, 201, 204, and 230; Class 709, subclasses 217 and 223; and Class 711, subclasses 162 and 202. Additionally, a literature search was conducted on the Internet for relevant non-patent documents and a search for

foreign patent documents was conducted on the ESPACENET and Delphion® databases.

(D) DOCUMENTS DEVELOPED BY THE PRE-EXAMINATION SEARCH AND OTHER ART OF RECORD IN THE APPLICATION

The documents located by the pre-examination search are listed below.

These documents were made of record in the present application by the Information Disclosure Statement filed September 9, 2004.

<u>Document No.</u>	<u>Inventor</u>
US 6,157,991	Arnon
US 6,260,124	Crockett et al.
US 6,353,878	Dunham
US 6,366,987	Tzelnic et al.
US 6,408,370	Yamamoto et al.
US 6,581,143	Gagne et al.
US 6,647,474	Yanai et al.
US 6,658,542	Beardsley et al.
US 20030051111	Nakano et al.
US 20030177321	Watanabe
US 20030188116	Suzuki et al.
US 20040024975	Morishita et al.
US 20040128442	Hinshaw et al.
US 20040148477	Cochran

Non-Patent Literature:

Role of Backup in Data Recovery, <http://www.storage.com>

Additionally, the following document was made of record in the present application by the Information Disclosure Statement filed May 12, 2005.

Non-Patent Literature:

SDRF/Asynchronous: A Technical Description, EMC White Paper, EMC Corporation, Feb. 2004.

Additionally, the following documents were made of record in the present application by the Information Disclosure Statement filed March 10, 2004.

<u>Document No.</u>	<u>Inventor</u>
US 6,092,066	Ofek
US 6,209,002	Gagne et al.
US 6,665,781	Suzuki et al.
EP 0 672 985	Kern et al.
EP 1 150 210	Tabuchi et al.

Because all of the above-listed documents are already of record in the present application, in accordance with MPEP § 708.02(VIII)(D), additional copies of these documents have not been submitted with this Petition.

(E) DETAILED DISCUSSION OF THE REFERENCES

Those of the above-listed documents deemed to be most closely-related to the present matter encompassed by the claims are discussed below in section 2, pointing out, with the particularity required by 37 CFR 1.111 (b) and (c), how the claimed present matter is patentable over the teachings of these documents.

1. Discussion of the Invention

The present invention teaches a system for maintaining consistency of data copied from a first storage system when the copied data is stored in a separate storage system. As set forth in claim 1, a feature of the invention is that if the write time is included in the write request, the first storage system records the write time and sends the write data and the write time to the second storage system, while if

the write time is not included in the write request, the first storage system assigns a write time recorded by the first storage system to the received write data, and then sends the write time and the write data to the second storage system for storage based on the write time. Similarly, under independent claim 7, the second storage system, instead of the first storage system, executes a similar process, and sends the write time and write data to a third storage system for storage in accordance with the write time. Accordingly, claims 1 and 7 are patentable because the prior art does not teach a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein in accordance with the write time.

2. Discussion of the References Deemed to be Most-Closely Related

Arnon, US 6,157,991, discloses a system for asynchronously updating a mirror storage of a source device. A source storage system 6 does not overwrite any data that has not yet been committed to a target storage system 8. The source storage system 6 maintains a data structure that includes information indicating which units of information updated in the source storage system have not yet been committed to the target storage system 8. In one embodiment of the invention, this information is maintained for each logical track. A data structure can be implemented in any of a number of ways. For example, a table can be provided in the global memory 25, and can be accessed by the host bus controller 19 prior to

executing any write command from the host CPU 1. When the host bus controller 19 determines that a command issued from the host CPU 1 is attempting to overwrite a logical track that has not yet been committed to the target storage system 8, the host bus controller prevents that write from occurring. (See, e.g., column 8, lines 26-47.) Thus, while Arnon discloses a system for asynchronously updating a mirror storage, in Arnon, a write time is not assigned to write data by a storage system. Accordingly, Arnon does not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein, as set forth in claims 1 and 7.

Crockett, et al., US 6,260,124, disclose a system for dynamically resynchronizing backup data that includes an update map and a progress queue. For resynchronization, once an update data record is represented in the progress queue, a comparison is made between the data record's read time-stamp (the time the data record was read from primary storage) and its write time-stamp (the time when a host originally sent the data record to the primary controller for writing). If the write time-stamp is earlier than the read time-stamp, the update will already be included in a static resynchronization. On the other hand, if the write time-stamp is later than the read time-stamp, this is a new update not included in the static resynchronization, and, thus, the dynamic resynchronization process applies it to the backup storage. (See, e.g., column 3, lines 1-10.) Accordingly, while Crockett et al.

teach a method for resynchronizing synchronous data backup, Crockett et al. fail to disclose a first or second storage system that assigns a write time to write data for enabling data transfer when a host computer does not apply a write time to the write data. Thus, Crockett et al. do not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein in accordance with the write time, as set forth in claims 1 and 7.

Dunham, US 6,353,878, and Tzelnic, et al., US 6,366,987, disclose similar systems for backing up data remotely in which the data is not backed up until a backup command is issued by a host 20. In response to a backup command from host 20, a primary data storage subsystem 21 accesses a primary directory 26 to find data of the physical storage unit specified by a backup command in order to initiate a process of copying data from a primary storage 27 to a secondary storage 29 of a secondary data storage subsystem 22. The primary data storage subsystem 21 creates an "instant snapshot copy" of the specified physical storage unit, and this instant snapshot copy is protected from modification by the host 20 while the instant snapshot copy is being written to the secondary storage 29. It is possible for the secondary storage 29 to contain more than one version of backup data for the same physical storage unit. In order to distinguish between different versions of backup data for the same physical storage unit, the primary data storage subsystem 21 appends an identification tag to the backup data transmitted from the primary data

storage subsystem to the secondary data storage subsystem 22. The tag, for example, is supplied by the host 20 in the backup command transmitted by the host to the primary data storage subsystem 21. The tag could also include a date-time stamp generated by the primary data storage subsystem. In the secondary data storage subsystem 22, the tag associated with each version of backup data is stored in a secondary directory 28, which further includes a record of a set of locations of the secondary storage 29 in which the version of backup data is stored. The tag associated with each version of backup data in the secondary storage 29 is used in a restore operation initiated by the backup software 24 in response to the user 23 or in response to a call from an application program executed by the host 20. (See, e.g., Dunham, column 5, line 28, through column 6, line 3.) Thus, Dunham and Tzelnic do not disclose that a storage system assigns a write time to write data. Accordingly, Dunham and Tzelnic do not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein, as set forth in claims 1 and 7.

Yamamoto, et al., US 6,408,370, disclose a host computer that gives a write time to each write data record when it issues a request for write to a primary controller. When the primary controller receives the write data from the host computer, it reports the completion of the receipt to the host computer. Then, the primary controller sends the write data records and the write times to the secondary

controller. At this time, the primary controller operates to send the write data records to the secondary controller in the sequence of the writing times. The secondary controller operates to store the write data received from the primary controller onto a non-volatile cache memory. This makes it possible to guarantee the write data without any I/O process of control information to and from disks. The secondary controller can guarantee the write data up to a certain time by referring to the received write time. (See, e.g., column 2, line 56 – column 3, line 7). Thus, Yamamoto et al. do not teach that a storage system assigns a write time to write data received from a plurality of computers, as in the present invention. Accordingly, Yamamoto et al. do not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein, as set forth in claims 1 and 7.

Gagne, et al., US 6,581,143, disclose a multiprocessor data processing system that includes a data storage facility wherein one program operates with data in one data storage device and a plurality of other programs, wherein each other program interacts with another data storage device. Multiple copies of the data from the one storage device are made on each of the additional data storage devices for operation with their corresponding programs. The interaction between these devices includes defining a first buffer for each additional storage device on which a copy is to be made and a second buffer for each additional storage device and the one

storage device. Data from the one storage device is copied to one of the additional data storage devices thereby to enable another program to interact with the data copy on the additional data storage device independently of the data and of the program being utilized with the one data storage device. Each change made by the one program to data on the one storage device and by the other program to the corresponding additional storage device is recorded in the first and second buffers, respectively. Upon completion of independent operation, the information in the corresponding first and second buffers can be combined to identify data to be copied from the one data storage device to one additional storage device thereby to enable the data to be copied so the data in the additional storage device replicates the data in the one data storage device. Thus, unlike the present invention, in Gagne et al. a write time is not assigned to write data by a storage system that then sends the write data and write time to another storage system. Accordingly, Gagne et al. do not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein, as set forth in claims 1 and 7.

Yanai, et al., US 6,647,474, disclose a data storage system for controlling the transfer or copying of data from a primary data storage system to a secondary data storage system, independent of and without intervention from one or more host computers. The data mirroring or copying is performed asynchronously with

input/output requests from a host computer. Accordingly, since data will not be immediately synchronized between the primary and secondary data storage systems, data integrity is maintained by maintaining an index or list of various criteria including a list of data which has not been mirrored or copied, data storage locations for which a reformat operation is pending, a list of invalid data storage device locations or tracks, whether a given device is ready, or whether a device is write-disabled. Information is included as to the time of the last operation so that the data may later be synchronized should an error be detected. Both the primary or secondary data storage systems maintain a table of the validity of data in the other storage system. The service processors in one embodiment periodically scan the index table for write-pending indicator bits and invoke a copy task which copies the data from the primary data storage device to the secondary. In addition, one or more of the spare index or table bits 114, 116 may be utilized to store other data such as time stamp, etc. (See, e.g., column 7, line 16, through column 8, line 7.) However, in Yanai et al. an index is maintained of data which has not been copied, whereas in the present invention, a write time is assigned to write data and sent with the write data to the other storage system. Accordingly, Yanai et al. do not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein, as set forth in claims 1 and 7.

Beardsley, et al., US 6,658,542, disclose a system and method for caching data. A processor receives data from a host to modify a track in a first storage device. The processor stores a copy of the modified data in a cache and indicates in a second storage device the tracks for which there is modified data in cache. During data recovery operations, the processor processes the second storage device and data therein to determine the tracks for which there was modified data in cache. The processor then marks the determined tracks as failed to prevent data at the determined tracks in the first storage device from being returned in response to a read request until the failure is resolved. (See, e.g., Abstract and column 5, line 55, through column 8, line 27.) Thus, Beardsley et al. do not teach the present invention, in which a storage system assigns a write time to write data. Accordingly, Beardsley et al. do not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein, as set forth in claims 1 and 7.

Nakano et al., US20030051111, disclose a remote copy control method and large area data storage system. Two data centers are connected using a synchronous transfer copy function, and one of the data centers is coupled with a third data center disposed at a remote location by an asynchronous remote copying function. The order in which a storage sub-system located in the vicinity has received data from a host is consistently guaranteed, and the third data center holds

the data. Further, each storage sub-system includes a function whereby, during normal operation, data can be exchanged and the data update state can be obtained by the storage sub-systems located in the two data centers that do not directly engage in data transmission. Thus, in Nakano, three or more data centers are interconnected by a transfer path along which data can be transmitted synchronously and asynchronously. Data update state management means is provided for each storage sub-system, and in order to cope with the occurrence of a disaster that can not be predicted, the update state management means appropriately monitors the data update state of a storage sub-system that is located in another data center, and transmits notification of the data update state of the storage sub-system to the others. Each of the storage sub-systems that do not directly engage in the transfer of data has a transfer state/bit map. To ascertain how many times and at which location in a transfer block data has been updated, one storage sub-system transmits inquiries that the other storage sub-system responds to. Thus, a function for monitoring and managing the state of data updating in remote copying is taught by Nakano et al. However, Nakano et al. fail to teach the present invention, such as a storage system that assigns write times to write data and sends the write data and the write time to a second storage system. Accordingly, Nakano et al. do not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein, as set forth in claims 1 and 7.

Watanabe, US20030177321, discloses data synchronization of multiple remote storage devices after remote copy suspension. Each of the storage systems includes volumes for storing data, and the volumes are maintained in a mirrored condition. If there is an interruption in the transmission of data between volumes, for example, caused by a failure of one or the other of the storage volumes, or a failure of the interconnecting network, a time-stamped bitmap created at the primary storage system is stored in one of the secondary storage subsystems. These records are then used to resynchronize the pair after the connection link is established. In the event that one member or the other of the pair fails, at the time of failure a record is made on a different storage volume of the status of the write operations to the failed storage volume. This record can then be used to resynchronize the storage volumes at a later time. (See, e.g., paragraphs 6-9 and 28-34.) Thus, Watanabe relies on a bitmap for resynchronization and, unlike the present invention, does not assign a write time to write data if a write time has not been assigned, wherein the write data and the write time are sent to another storage system, and the write data is stored based on the write time. Accordingly, Watanabe does not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein in accordance with the write time, as set forth in claims 1 and 7.

Suzuki et al., US20030188116, disclose a method and apparatus for managing backup and restoring among copy volumes in a storage system. The storage system includes a plurality of copy volumes, with each of the copy volumes originating from a single volume and configuring a pair with the single volume. Each of the copy volumes is capable of being independently used. Under the managing method of Suzuki et al., when the volumes configuring each pair are used by separate applications, for each pair in a memory, an address is stored at which updating of data has been made in the volumes as updated-location-managing information. In case a copy volume configuring a first pair is to be restored to have the contents of a copy volume belonging to a second pair, pieces of the updated-location-managing information for each of the first and second pairs are obtained from memory. Differential information is generated by merging the pieces of updated-location-managing information obtained. The differential information is indicative of a difference in the pieces of updated-location-managing information between the copy volumes. The copy volume which is to be restored is reproduced to have the contents of the copy volume to be the source of restoring, by copying data designated by the differential information from the copy volume to be the source of restoring to the copy volume which is to be restored. (See, e.g., paragraph 9.) Thus, Suzuki et al., do not teach the present invention, including a storage system that assigns a write time to write data received from a plurality of computers. Accordingly, Suzuki et al. do not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been

assigned, and sends the write data with the write time to another storage system for storage therein, as set forth in claims 1 and 7.

Morishita et al., US 20040024975, disclose a storage system for multi-site remote copy. A sequence number counted is included for recording the write order from a host. The transfer-target management information is information by which the order in which data will be transferred to a copy destination is managed in a copy source of asynchronous remote copy. Transfer-target management information is used as a queue header of a queue structure formed out of sequence management information entries. First, the storage system CPU 30 secures a sequence management information entry for data newly written. Next, when the write source of the data is the host, the sequence number of the written data is acquired from a sequence number counter. Alternatively, when the write source of the data is another storage system 2, the sequence number included in the received data is acquired. Then, the CPU 30 writes the acquired sequence number into the secured sequence management information entry. When the write source of the data is the host, the CPU 30 updates the value of the sequence number counter after acquiring the sequence number. (See, e.g., paragraphs 44-47.) Thus, Morishita uses an assigned sequence number for formalizing data, and does not include a storage system that assigns a write time to write data received from a plurality of computers, as in the present invention. Accordingly, Morishita et al. do not teach or suggest a storage system that receives write data, assigns a write time to the write data if a

write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein, as set forth in claims 1 and 7.

Further, Morishita et al. is owned by the Assignee of the present application, and the restrictions of 35 USC 103(c) apply when assessing patentability.

Suzuki et al., US 6,665,781, discloses a method for duplexing data in a storage unit system to carry out backup operation of data from a primary-side system having one or more primary-side control units to a secondary-side system having one or more secondary-side control units. Under the method, time information relating to a write time is added to a data of a write request. The information is stored in a cache memory in the primary-side control unit when there is a write request from a processing unit to the primary-side control unit. The stored write data and the time information is transmitted to the secondary-side control unit. The write data and the time information is stored to a cache memory in the secondary-side control unit. The time information stored in the cache memory in the secondary-side control unit is transmitted to another secondary-side control unit via a communication route connecting the secondary-side control units with each other. The time information of a first secondary-side control unit received from a first primary-side control unit is compared with a time information transmitted from a second-secondary side control unit. An older time information is transmitted to the next secondary side control unit via the communication route. The older time information is circulated between the plural secondary side control units, and a data guarantee

time is determined for guaranteeing duplexing of the write data between the plural secondary side control units. Thus, while Suzuki et al. disclose the use of time information with write data, they do not disclose that a second or third storage system stores write data based on or in accordance with a write time assigned to the write data by a first or second storage system, respectively, as recited by the present claims 1 and 7. Accordingly, Suzuki et al. do not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein in accordance with the write time, as set forth in claims 1 and 7.

Kern et al., EP 0 672 985, is also discussed at pages 1-4 of the specification of the present application, and discloses a primary storage subsystem that is synchronized by a common timer, and a secondary system, remote from the primary processor, which shadows the data updates in sequence-consistent order such that the secondary site is available for disaster recovery purposes. Under the method of Kern, time-stamping of each write I/O operation occurs in the primary storage subsystem. Write I/O operation record set information is written from the primary storage subsystem for each data update. Self-describing record sets are generated from the data updates and the respective record set information, such that the self describing record sets are sufficient to re-create a sequence of the write I/O operations. The self-describing record sets are grouped into interval groups based

upon a predetermined interval threshold. A first consistency group is selected as that interval group of self-describing record sets having an earliest operational time stamp, with the individual data updates being ordered within the first consistency group based upon time sequences of the I/O write operations in the primary storage subsystem. Thus, Kern et al. disclose time-stamping of each write I/O operation in the primary storage subsystem, but Kern et al. do not teach the present invention in which the second storage system stores write data based on or in accordance with write times assigned to the write data. Accordingly, Kern et al. do not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein in accordance with the write time, as set forth in claims 1 and 7.

Tabuchi et al., EP 1 150 210, discloses a method of duplicating data of a system which is provided with a first storage subsystem group comprising two or more storage subsystems and a second storage subsystem group comprising two or more storage subsystems which store copies of the data of the first storage subsystem group. The data written into each of the storage devices of the storage subsystems which belong to the first storage subsystem group are given serial numbers and times. The data is transferred to the storage subsystems which belong to the second storage subsystem group. The two or more data received by each of the storage subsystems which belong to the second storage subsystem group are

arranged in sequence of the serial numbers, the oldest time is decided comparing the latest times given to each of the storage subsystems by the communication among the storage subsystems which belong to the second storage subsystem group, and the data with the times earlier than the decided oldest time are the objects of data writing into the storage devices of the storage subsystems. Thus, Tabuchi et al. use a comparison technique of oldest times for determining which data to store, but Tabuchi et al. do not disclose the present invention, including a system in which a second or third storage system stores write data based on or in accordance with time information assigned to the write data by a first or second storage system, respectively, and which is then sent with the write data. Accordingly, Tabuchi et al. do not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein in accordance with the write time, as set forth in claims 1 and 7.

3. Remaining References

The remaining references of record in the application are deemed to be less-closely related to the present invention, and/or were provided as background information, and also do not show or suggest the present invention. However, a discussion of each of these remaining references has been provided to avoid dismissal of the Petition in case the Examiner disagrees with Applicants'

determination that these references are not most-closely related to the subject matter of the claims.

The EMC White Paper, *SDRF/Asynchronous: A Technical Description*, has a publication date of February 2004, while the present application has a US filing date of March 10, 2004, and a foreign priority date of December 3, 2003. Accordingly, this document does not qualify as prior art under US law for purposes of assessing the patentability of the present claims, and no further discussion of this document is required for this Petition.

Hinshaw et al., US20040128442, disclose a disk mirror architecture that may include a system manager, which may create a spare processing unit by redistributing data stored on a processing unit that is actively in use, among a subset of the plurality of other processing units. The system manager redistributes the data by reassigning blocks in a distribution map. The system manager also includes a background task that performs the redistribution of the data. The system manager also monitors all processing units and redistribute data among the disks coupled to the processing units upon detecting a change in network topology. The distribution map stores a striping scheme for a table. The striping scheme may distribute the table across a plurality of disks or may stripe the table on one of the plurality of disks. (See, e.g., paragraphs 6-14 and 70-80.) Thus, the disclosure of Hinshaw et al. is not directed to the same subject matter as the present invention, i.e., storing copy data

between storage systems; and Hinshaw et al. do not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein, as set forth in claims 1 and 7.

Cochran, US20040148477, discloses a method and system for providing logically consistent logical unit (LUN) backup snapshots within one or more data storage devices. Backup logical units within a pool are continuously recycled, so that the least-recently-current backup logical unit is next synchronized and activated to receive mirror I/O requests. A trigger I/O request is provided to allow an application program running on a host computer to signal points within a stream of I/O requests that represent logically consistent states. The controller of a data storage device recognizes a special I/O request received from a host computer as a TRIGGER event, and inserts a corresponding TRIGGER message into a sequenced stream of I/O requests for a particular primary LUN of a primary LUN/backup LUN mirror pair. The TRIGGER message indicates to the data storage device controller that, when all I/O requests preceding the TRIGGER message are successfully executed on the primary LUN, the primary LUN will be in a logically consistent state. The TRIGGER message can be, in turn, inserted into the I/O request stream directed to the backup LUN, so that the backup LUN can also detect a logically consistent state. Thus, Cochran does not teach a storage system that assigns time information to write data. Accordingly, Cochran does not teach or suggest a storage system that receives write

data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein, as set forth in claims 1 and 7.

"Role of Backup in Data Recovery", (<http://www.storage.com>) discusses data backup issues in general. Discussed are factors impacting backup, conventional tape backup in today's market, issues with today's backup, and the requirement for new architectures and techniques. The document states that mirroring, or live data replication for "hot" recovery has a role in systems where data must be always available. Remote hot recovery sites are needed for immediate resumption of data access. Further, the document states that the primary storage device must employ memory mapping techniques that enable the tracking of all modified information, with two copies of each change being kept, with a thread composed of all old data stored in a journaled file. The document advocates putting mechanisms into place to allow the backup of data to occur directly from the primary storage area to the backup area without intervention from the host. Thus, this document is only of general interest, and does not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein, as set forth in claims 1 and 7.

Ofek, US 6,092,066, is also discussed at pages 3-4 of the specification of the present application, and discloses a data processing system that includes redundant storage of data and that enables access to the data by multiple processes. A communications link interconnects first and second data processing systems, each being capable of independent operation and including a host computer and data storage facility that stores a data collection at predetermined locations in data blocks. During a normal operating mode, the second system mirrors the data in the first system data storage facility. The second system can operate in an independent operating mode by disabling transfers through the communications link. While communications are disabled, the first system records an identification of each data block that it alters in its data storage facility. The second system records an identification of each data block in its data storage means that changes as a result of its operation. When the independent operation of the second system terminates, the communications link re-enables transfers. Data blocks with combined recorded identifications are copied from the first to the second system data storage facilities to reestablish the second data processing system data storage facility as a mirror of the first data processing system storage facility. (See, e.g., column 3, line 47, through column 4, line 5.) Thus, Ofek is directed to resynchronizing a mirroring copy system following a disablement of communications, and does not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein, as set forth in claims 1 and 7.

Gagne et al., US 6,209,002, is also discussed at pages 3-4 of the specification of the present application, and discloses a storage system in which data is transferred between a production site and a remote site. The production site includes a host and production storage facility. The remote site is a remote storage facility, including a first data store and a second data store wherein the first data store receives data from the production facility on a track-by-track basis. The host can issue a plurality of cascade commands to enable the definition of a plurality of track status tables for identifying each track in the first data store that the production facility changes. In addition, the host can establish first and second operating modes. In the first operating mode, the second data store receives data from the first data store according to the changes recorded in the track status tables. During the second operating mode data from the second data store is copied to the remote storage facility according to the changes recorded in the track status tables at the time the second mode is established. (See, e.g., column 2, line 65, through column 3, line 12.) Thus, Gagne et al. do not teach or suggest a storage system that receives write data, assigns a write time to the write data if a write time has not already been assigned, and sends the write data with the write time to another storage system for storage therein, as set forth in claims 1 and 7.

CONCLUSION

Thus, from the foregoing, it is apparent that none of the above-listed documents teach or suggest a first storage system in which, if the write time is included in the write request, the first storage system records the write time and sends the write data and the write time to the second storage system, and if the write time is not included in the write request, the first storage system assigns a write time recorded by the first storage system to the received write data and then sends the write time and the write data to the second storage system for storage based on the write time. Nor do the above-listed documents teach or suggest a first storage system that stores write data received from a plurality of computers and sends the write data received from the plurality of computers to a second storage system, wherein if a write time is included in the write request, the second storage system records the write time and sends the write data and the write time to a third storage system, and if the write time is not included in the write request, the second storage system assigns the write time to the received write data and then sends the write time and the write data to the third storage system for storage in accordance with the write time, as set forth in claim 7. Accordingly, independent claims 1 and 7 are patentable over the above-discussed documents.

The Applicants submit that the foregoing discussion demonstrates the patentability of the independent claims over the closest-known prior art, taken either singly, or in combination. The remaining claims depend from the independent

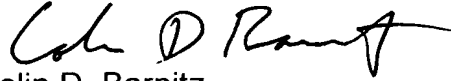
claims, claim additional features of the invention, and are patentable at least because they depend from allowable base claims. Accordingly, the requirements of 37 CFR §1.102(d) having been satisfied, the Applicants request that this Petition to Make Special be granted and that the application be examined according to prescribed procedures set forth in MPEP §708.02 (VIII).

The Applicants prepared this Petition in order to satisfy the requirements of 37 C.F.R. §1.102(d) and MPEP §708.02 (VIII). The pre-examination search required by these sections was "directed to the invention as claimed in the application for which special status is requested." MPEP §708.02 (VIII). The search performed in support of this Petition is believed to be in full compliance with the requirements of MPEP §708.02 (VIII); however, Applicants make no representation that the search covered every conceivable search area that might contain relevant prior art. It is always possible that prior art of greater relevance to the claims may exist. The Applicants urge the Examiner to conduct his or her own complete search of the prior art, and to thoroughly examine this application in view of the prior art cited above and any other prior art that may be located by the Examiner's independent search.

Further, while the Applicants have identified and discussed certain portions of each cited reference in order to satisfy the requirement for a "detailed discussion of the references, which discussion points out, with the particularity required by 37 C.F.R. §1.111(b) and (c), how the claimed present matter is patentable over the references" (MPEP §708.02(VIII)), the Examiner should not limit review of these

documents to the identified portions, but rather is urged to review and consider the entirety of each reference.

Respectfully submitted,



Colin D. Barnitz
Registration No. 35,061
Attorney for Applicants

MATTINGLY, STANGER, MALUR & BRUNDIDGE, P.C.
1800 Diagonal Rd., Suite 370
Alexandria, Virginia 22314
(703) 684-1120
Date: May 12, 2005